

Spectral Graph Analysis for Process Monitoring

Estanislao Musulin*

*Centro Internacional Franco Argentino de Ciencias de la Información y de Sistemas,
Ocampo y Esmeralda, S2000BTP, Rosario, Argentina*

E-mail: musulin@cifasis-conicet.gov.ar

Abstract

Process monitoring is a fundamental task to support operator decisions under abnormal situations. Most process monitoring approaches, such as Principal Components Analysis and Locality Preserving Projections, are based on dimensionality reduction. In this paper Spectral Graph Analysis Monitoring (SGAM) is introduced. SGAM is a new process monitoring technique that does not require dimensionality reduction techniques. The approach it is based on the spectral graph analysis theory. Firstly, a weighted graph representation of process measurements is developed. Secondly, the process behavior is parameterized by means of graph spectral features, in particular the graph *algebraic connectivity* and the graph *spectral energy*. The developed methodology has been illustrated in autocorrelated and non-linear synthetic cases, and applied to the well known Tennessee Eastman process benchmark with promising results.

1 Introduction

In complex modern processes, the interest on monitoring systems has increased following the demand of better management of the plants according to more restrictive economical and

*To whom correspondence should be addressed

environmental conditions. In the last years, many process monitoring methods have been developed to timely detect process disturbances.¹⁻⁴ In particular data driven techniques have attracted the greatest interest due to the high availability of on-line data.⁵⁻⁹ Many process monitoring applications relies on dimensionality reduction techniques; process measurements are projected into a low-dimension space where most of the normal data variability is contained. Therefore, the monitoring performance is directly influenced by the quality of the projection model. The most widespread of these techniques is the Principal Components Analysis (PCA)^{10,11} and its extensions.^{3,12-15} PCA examines the measurements covariance and selects a new orthogonal base of reduced dimensionality that explains most of the data variance. Thus, PCA can be considered as a globality-based data projection method that does not consider the local structure of data. That is, data neighborhood can be altered after projection.

Recently, several methods of dimensionality reduction where proposed in the pattern recognition area,¹⁶⁻²⁰ which are reviewed and systematically compared in.²¹ As opposed to PCA, those methods, known as *manifold learning*, are based on the local structure of data. Moreover, most of these approaches are non-linear and computationally expensive. Between them, Locality Preserving Projections (LPP)¹⁹ is of particular interest; LPP is a linear technique that defines the neighborhood relationship between data samples and finds the projection that preserves the intrinsic geometry structure of the dataset. However, the outer shape of the dataset can be modified and therefore the global data structure can be distorted.

Lately, Global-Local Structure Analysis (GLSA)²² and Local and Global PCA (LGPCA)²³ were proposed as compromise solutions. They construct dual objective functions aiming to preserve both local and global dataset structure.

In this work an alternative process monitoring technique based in Spectral Graph Analysis (SGA) is proposed. As in manifold learning techniques, a similarity graph is constructed based on the normal dataset, however, instead of using this information to project data

into a dimensionality reduced space, the monitoring is performed by directly studying the spectral characteristics of the graph. Before going into further details, some general graph concepts will be introduced.

Many real-world situations can conveniently be represented by a diagram consisting of a set of vertices joined by a set of edges. For example, vertices could be people, communication centers or process variables, and the corresponding edges could represent friendship links, communications links and interactions degree, respectively. A mathematical abstraction of these situations leads to the concept of graph.²⁴ A graph can be represented by its adjacency matrix \mathbf{A} ; the analysis of \mathbf{A} based on eigenvalues and eigenvectors is called the theory of graph spectra.^{25,26} This theory attempts to utilize linear algebra, including the well-developed theory of matrices, for purposes of graph theory and its applications.²⁷

Spectral graph theory has acquired great relevance in the last decade, particularly in the computer science area. An excellent survey of these applications can be found in.²⁸ Other research areas such as process engineering have very few applications of this spectral approach. Anyway, there are some works related to graph theory such as²⁹ where new methods for the analysis of complex processes are suggested. The authors formulated a flexible framework to help the designers in comprehending a process by representing structural and functional relationships. Yang and co-workers³⁰ suggested a fusion of information from process data and process connectivity. Signed directed graphs are used to capture the process topology and connectivity, thus depicting the causal relationships between process variables. Gutierrez-Perez and co-workers³¹ introduced a methodology based on spectral measurements of graphs to establish the relative importance of areas in water supply networks. These areas are analyzed using a flexible method of semi-supervised clustering. Finally Zumoffen and Musulin³² build a graph using the process static sensitivity matrix, and show the utility of a spectral analysis to optimally select the variables to be controlled.

In this work, SGAM is introduced as an alternative method for process disturbance detection. It is shown that processes can be monitored by analyzing the spectral features of

a properly defined weighted graph.

The paper is organized as follows. Section 2 gives some insight into spectral graph theory with particular reference to some results that motivated this work. In section 3 the new monitoring methodology is presented. Experimental results are shown in section 4. Finally in section 5 conclusions are given and some future works are proposed.

2 Spectral Graph Analysis

Let $G = (V, w)$ be a finite undirected weighted graph of order n without loops or multiple edges, and suppose that its vertices are labeled $1, 2, \dots, n$. If vertices i and j are joined by an edge, it is said that i and j are *adjacent* and write $i \sim j$. Since the graph is weighted and undirected, it is assumed that each edge carries a non-zero symmetric weight ($w_{ij} = w_{ji}$). The elements of the *weighted adjacency matrix* \mathbf{A}_w of the weighted graph G are defined as²⁶

$$a_{ij} = \begin{cases} w_{ij} \neq 0 & \text{if } i \sim j \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Under this definition, \mathbf{A}_w is a real symmetric matrix with zero diagonal.

2.1 Graph Energy

The spectrum (eigenvalues of \mathbf{A}_w) is a graph invariant. Let $\lambda_1, \dots, \lambda_n$ be such eigenvalues of \mathbf{A}_w . Then, the energy of G is defined as^{26,33}

$$\mathbb{E}(G) = \sum_{i=1}^n |\lambda_i| \quad (2)$$

This quantity is well known in chemical applications; since in some cases the energy defined in this way corresponds to the energy of a molecule.^{27,34} However, the graph invariant $\mathbb{E}(G)$ can be considered for any graph independently of the chemical context, recently

much work on graph energy appeared also in the pure mathematics literature.^{26,35} This new perspective provided new generally valid mathematical properties for $\mathbb{E}(G)$.

In this paper is of particular importance the result obtained by Gutman and Shao²⁶,

Theorem 1. *Let $G_1 = (G, w_1)$ and $G_2 = (G, w_2)$ be two weighted graphs on a graph G . Suppose that there is an edge $e_0 = ij$ of G , satisfying the following conditions:*

- $w_1(e) = w_2(e)$ for all $e \in \mathbb{E}(G)$ with $e \neq e_0$;
- $w_2(e_0) > w_1(e_0) > 0$;
- $\mathbb{E}(G_1 - e_0) < \mathbb{E}(G_1)$ where $G_1 - e_0$ is the weighted graph obtained from G_1 by deleting the edge e_0 . Then, $\mathbb{E}(G_2) > \mathbb{E}(G_1)$.

It says that if removing some edge e_0 from a weighted graph G decreases the energy, then increasing the positive weight on that edge will increase the energy.

2.2 Algebraic connectivity

Given \mathbf{A}_w , the discrete Laplacian matrix \mathbf{L} is defined as $\mathbf{L} = \mathbf{A}_w - \mathbf{D}$, where \mathbf{D} is the diagonal matrix of vertex degrees $\mathbf{D} = \text{diag}(a_i)$, and $[\mathbf{D}]_{ii} = a_i = \sum_{j=1}^n a_{ij}$. \mathbf{L} is a positive semi-definite matrix and so all its eigenvalues are no negative. The second smallest eigenvalue of \mathbf{L} is usually called *algebraic connectivity* of G ²⁷ and it is denoted by $a(G)$.

Although a simple eigenvalue is not enough to describe the graph structure, the algebraic connectivity have some important relations with several graph metrics. As a consequence, it has been used in separation, metric and isoperimetric problems.²⁷ Our interest in $a(G)$ relies in the following inequalities,

Theorem 2.³⁶ *If G is a connected graph on n vertices, then*

$$\text{diam}(G) \geq \lceil \frac{4}{n a(G)} \rceil \quad (3)$$

Where $diam(G)$ is the graph diameter, defined as the maximum distance between any two vertices of G .

Theorem 3.³⁷ *For any graph G on n vertices ($n \geq 2$),*

$$i(G) \geq a(G)/2 \quad (4)$$

Where $i(G)$ is the *isoperimetric number* or *conductance*, defined as,²⁷

$$i(G) = \min_{0 < |S| \leq n/2} \frac{|\delta S|}{|S|} \quad (5)$$

where S is a subset of vertices of G , δS is the *edge boundary* of S and $|\cdot|$ is an operator that return the number of vertices or edges of a set.

If $i(G)$ is small, then a relative big number of vertices can be separated from the graph by removing relatively few edges (in particular $i(G) = 0 \iff G$ is disconnected). Therefore, $i(G)$ (and so $a(G)$) can be considered as a numerical measure of the graph conductance, and consequently as a measure of connectivity.

3 Use of SGA in process monitoring

Considering the above presented results (i.e. Theorems 1, 2 and 3), in this paper it is proposed to build a graph G that represents the last process measurements structure, and monitor the process by analyzing changes in the spectral features of G . In particular, the spectral energy $\mathbb{E}(G)$ and the algebraic connectivity $a(G)$ are studied.

Before presenting the algorithm details, in the next subsection a motivating example is shown.

3.1 Motivating example: the swiss-roll dataset

In this subsection spectral graph features are extracted from the swiss-roll dataset.²¹ Then, three different modifications (disturbances) have been introduced. The normal 3D dataset X has been generated using the following equations,

$$\begin{aligned} t &= \frac{3\pi}{2}(1 + 2d_1) \\ X &= [\frac{t \cos(t)}{5}, 6d_2 - 3, \frac{t \sin(t)}{5}] + 0.05N \end{aligned} \tag{6}$$

d_i are different realizations of a random vector $\mathcal{U}(0, 1)$ of dimension 1000, and N is a random matrix of dimension 3×1000 and distribution $\mathcal{N}(0, 1)$.

In order to simulate three different disturbances (IDV(1), IDV(2) and IDV(3), a series of points has been replaced in the dataset and abnormal ones have been introduced. For IDV(1) and IDV(2) two points out of the surface have been introduced: $p_1 = [0, 0, 0]$ and $p_2 = [-1, 0, 12/5]$ respectively.

For IDV(3), an accumulation of points in a small part of the surface has been simulated, $\mathbf{p}_3 = [(0.2\mathbf{d} + 12.4)/5, 2\mathbf{d} - 1, (2\mathbf{d} - 1)5]'$, where \mathbf{d} is a random vector of 100 samples with distribution $\mathcal{U}(0, 1)$.

A graph is built for each dataset and the modification on the spectral properties are shown and compared with the PCA and LPP projections.

To build a graph, a full adjacency matrix \mathbf{A}_w has been calculated on X using the following equation:

$$a_{ij} = w_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}} \tag{7}$$

where \mathbf{x}_i and \mathbf{x}_j are points in X and $\sigma = 1/5$ indicates the variance of the Gaussian (i.e. σ controls the width of the neighborhoods).

Based on a realization of the normal dataset (Figure 1a), PCA and LPP projections have been performed. Figure 1b and Figure 1c show the 2D projections using PCA and LPP

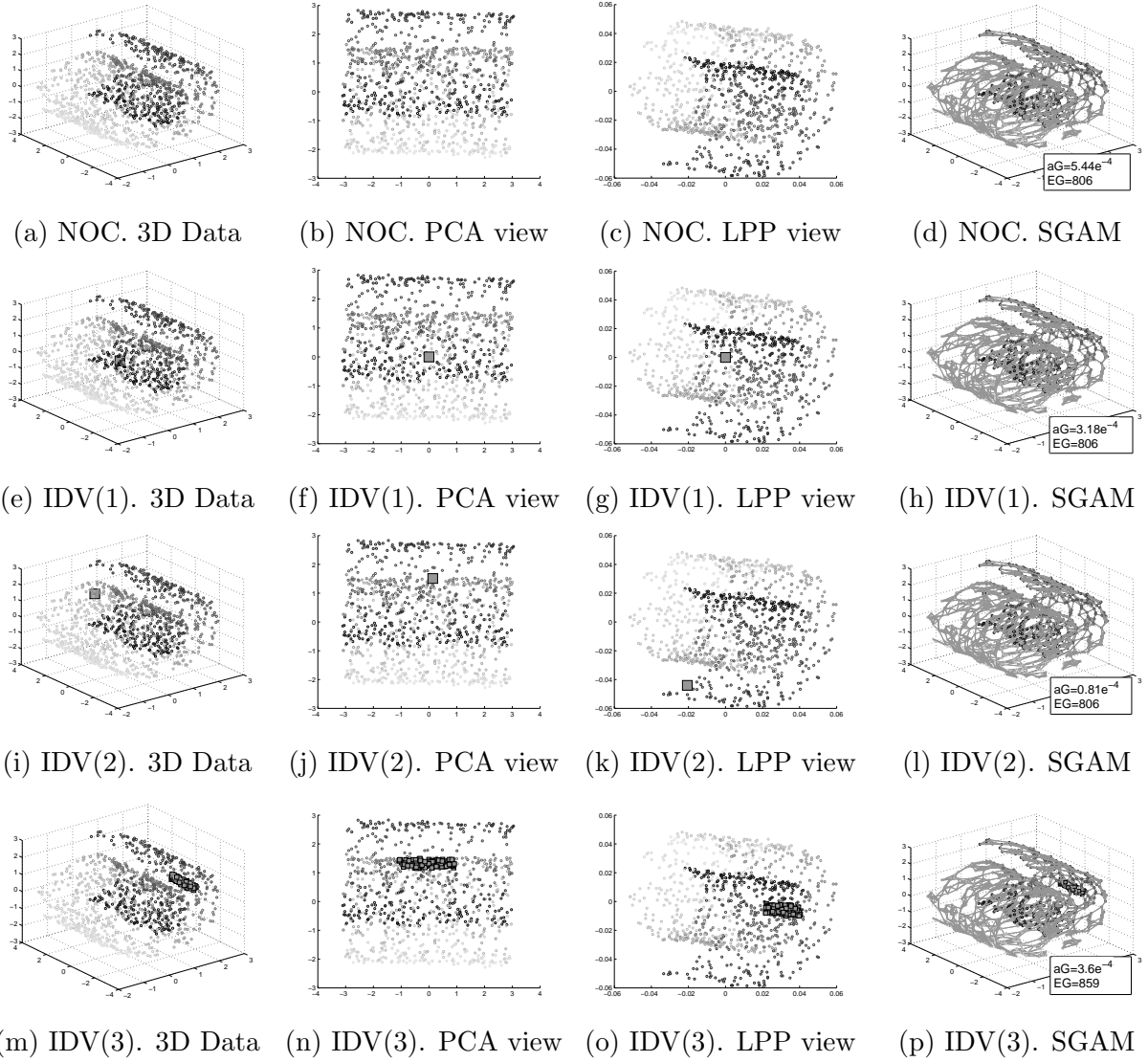


Figure 1: Swiss Roll dataset representation, abnormalities are depicted as square points. The first column shows the original data, column 2 and 3 depict the PCA and LPP projections respectively. Column 4 shows the SGAM graphs and their spectral features $a(G)$ and $\mathbb{E}(G)$. In the SGAM graphs only the five strongest connection have been depicted for each vertex.

respectively. Figure 1d, shows the corresponding graph and its spectral parameters $a(G)$ and $\mathbb{E}(G)$. To facilitate the view, only the five stronger connections of each vertex have been depicted. Note that no dimensionality reduction is performed in this case.

Figure 1e to 1p show the projection of these disturbances in the PCA and LPP reduced spaces and the changes in the graph spectral properties $\mathbb{E}(G)$ and $a(G)$. Note that these points can not be distinguished from the normal dataset in the LPP and PCA projections, but they cause clear changes in the graph spectral features. Points out of the mesh (p_1 and p_2) reduce the algebraic connectivity, while points localized in a reduced area (p_3) increase the energy of the graph. The latter is an important result in disturbance detection, because it shows that points do not need to be placed out of the manifold to be considered an abnormality by SGAM.

Bearing in mind these results, in the next subsection an algorithm for process monitoring is formally presented.

3.2 The SGAM algorithm

To perform on-line monitoring based on the spectral graph features, it is proposed the construction of a graph G in which each vertex represents a process measurement vector $\mathbf{x}_m(k) \in \mathbb{R}^m$.

If the normal process measurements are analyzed in groups of n samples, each group can be characterized as a graph of n disconnected vertices each one of dimension m . To establish connections, the graph adjacency matrix \mathbf{A}_w can be determined by calculating the Euclidean distances between vertices and applying a heat kernel (See Eq. 7) as a measure of neighborhood. Note that \mathbf{A}_w results of dimension $n \times n$ independently of the dimension of the measurement vector m .

To establish data groups, the Normal Operation Conditions (NOC) dataset is studied using a moving window ($TW_{n \times m}(k) = [\mathbf{x}_m(k-n), \dots, \mathbf{x}_m(k)]$), thus obtaining a series of graphs represented by their corresponding adjacency matrix \mathbf{A}_w . $\mathbb{E}(G)$ and $a(G)$ are cal-

culated for each \mathbf{A}_w as explained in section 2. The obtained $\mathbb{E}(G)$ and $a(G)$ sequences are considered as normal and disturbed only by common cause variance, so limits of normality can be estimated from them. This procedure is formally stated in Table 1.

Table 1: SGAM process characterization stage. Build graphs and Establish normal limits for $\mathbb{E}(G)$ and $a(G)$

Consider an auto-scaled (zero mean, unit variance) normal data matrix \mathbf{X} consisting of N samples of m variables, and so $\mathbf{X} = [\mathbf{x}_m(1), \mathbf{x}_m(2), \dots, \mathbf{x}_m(N)]$, with $\mathbf{x}_m(k) \in \mathbb{R}^m$.

1. Take a time windows $TW(n)$ consisting on the first n samples of \mathbf{X} (i.e. $TW(n) = [\mathbf{x}_m(1), \dots, \mathbf{x}_m(n)]$), where $n < N$.
 - 1a. Construct a weighted graph to represent $TW(n)$: Let G denote a graph with n vertices. Build a fully connected graph by simply connecting all points with positive similarity with each other. Weights are assigned to graph edges computing the Gaussian kernel function, leading to a matrix \mathbf{A}_w with entries $w_{ij} = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2\sigma^2}}$, where σ indicates the variance of the Gaussian (i.e. σ controls the width of the neighborhoods), and $\|\cdot\|$ indicates the Euclidean norm operator.
 - 1b. Calculate the energy of G as $\mathbb{E}(G) = \sum_{i=1}^n |\lambda_i|$, where λ_i are the eigenvalues of \mathbf{A}_w . Note that with this definition, G is a connected graph and so only the first Laplacian eigenvalue will be null with multiplicity one.²⁰
 - 1c. Calculate the graph algebraic connectivity $a(G)$ as explained in section 2.2. It requires to compute the graph Laplacian matrix $\mathbf{L} = \mathbf{A}_w - \mathbf{D}$ of dimension $n \times n$ and the calculation of its second smallest eigenvalue.
 2. Take a new time window $TW(n+1) = [\mathbf{x}_m(2), \dots, \mathbf{x}_m(n+1)]$. To analyze $TW(n+1)$ it is not necessary to rebuild all the graph, only the following steps are required,
 - 2a. Remove the first column of \mathbf{A}_w corresponding to the older point $\mathbf{x}_m(1)$.
 - 2b. Add $\mathbf{x}_m(n+1)$ to the graph by calculating the weighted distance to the other graph vertices, and update \mathbf{A}_w .
 - 2c. Recalculate $\mathbb{E}(G)$ and $a(G)$.
 3. Continue the procedure from step 2 until the complete normal dataset have been processed.
-

Using the obtained normality limits, two control charts are built: the *EG-control chart* and the *aG-control chart* to monitor $\mathbb{E}(G)$ and $a(G)$ respectively. Then, on-line monitoring is performed by analyzing the last n measurements in a sliding window, building G , calculating $\mathbb{E}(G)$ and $a(G)$ and comparing the obtained values with the established normality limits in the control charts. See Table 2 for a step by step explanation.

Table 2: SGAM on-line monitoring stage. Compute $\mathbb{E}(G)$ and $a(G)$ and compare with normality limits

-
1. Initialize \mathbf{A}_w
 - 1a. Take the last n measurements $TW(k) = \mathbf{x}_m(k-n), \dots, \mathbf{x}_m(k)$.
 - 1b. Build the weighted graph \mathbf{A}_w with entries $w_{ij} = e^{\frac{\|\mathbf{x}_m(i) - \mathbf{x}_m(j)\|^2}{2\sigma^2}}$
 - 1c. Calculate $\mathbb{E}(G)$ and $a(G)$, plot them in their corresponding control graph and compare with the normality limits.
 2. Until each new measurement $\mathbf{x}_m(k+1)$ Do:
 - 2a. Remove the first column of \mathbf{A}_w corresponding to the older point $\mathbf{x}_m(k-n)$.
 - 2b. Add $\mathbf{x}_m(k+1)$ to the graph by calculating the weighted distance to the other graph vertices, and update \mathbf{A}_w .
 - 2b. Recalculate $\mathbb{E}(G)$ and $a(G)$, plot them in their corresponding control graph and compare with the normality limits.
-

3.3 Selection of parameters

To apply the presented monitoring technique two parameters are needed, the Gaussian variance of the heat kernel (σ) and the number of samples in each time window (n).

The optimal values for these parameters are not easy to find. As a rule of thumb, if the input dataset is normalized with unit variance and assuming Gaussian distribution, most of the dataset will be distributed in a m-sphere of radio 5σ . Then, σ can be chosen between 1-5.

On the other hand, the size of the time window n must be big enough to explain most of the common cause variance, including the noise variability and the dynamics of the process.

The following procedure is proposed to make a more accurate estimation of σ .

Lets suppose that G has been calculated for K normal time windows and denote by $d_{max}(k)$, $k = 1, \dots, K$ the sequence of maximum distances between vertices in each graph. The normal dataset *radio* is defined as,

$$R_N = \text{mean}(d_{max}(k)), k = 1, \dots, K \quad (8)$$

Then σ it is chosen so that vertices (i,j) distanced more than R_N are considered disconnected (i.e. $w_{ij} \leq e^{-5}$), using Eq. 7,

$$w_{ij} = e^{-5} = e^{-\frac{R_N^2}{2\sigma^2}} \quad (9)$$

and hence,

$$\sigma = \frac{R_N}{\sqrt{10}} \quad (10)$$

which is the proposed estimation.

3.4 Computational complexity analysis

A key limitation in any on-line application are computation times. In the following, the efficiency of the SGAM algorithm is assessed both, theoretically and experimentally.

To update the moving window, the algorithm requires the consecutive addition of one vertex ($\mathbf{x}_m(k) \in \mathfrak{R}^m$), and the calculation of the distances from the new vertex to the $n - 1$ previous ones, so the complexity goes to $O(nm)$. On the other hand, the complexity to obtain the eigenvalues of \mathbf{A}_w and \mathbf{L} is at most $O(n^3)$.³⁸ Therefore, the total algorithm complexity results,

$$T = O(mn + 2n^3) \quad (11)$$

Note that this value depends mainly on the size of the moving window (n), which is a fix parameter, and only grows linearly with the number of variables (m).

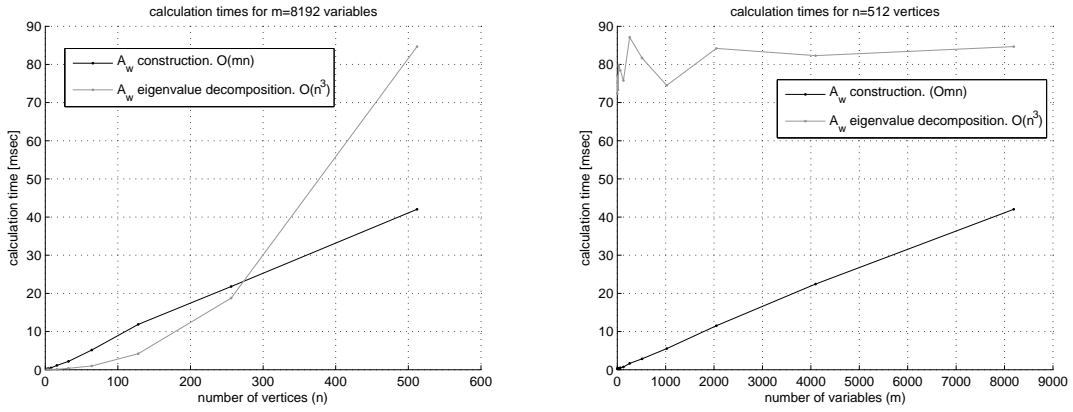
To experimentally evaluate the calculation performance, several runs of SGAM where performed using several combination of parameters (n and m), results are presented in figure 2. Note that these results are in accordance with the calculated complexity given in Eq. 11. All the experiments were run in Matlab, using an Intel i5-720 processor with 4GB of RAM.

For example, to illustrate this in a real case, consider a process with $m = 1000$ variables, and a monitoring time window of $n = 120$ vertices (note that if the process is sampled every

minute, $n = 120$ does imply a 2 hours monitoring time window). For this combination of parameters, calculation times result,

- Time to calculate distances and to build the adjacency matrix $\mathbf{A}_w = 1.5\text{ms}$
- Time to perform the eigenvalue decomposition of \mathbf{A}_w and $\mathbf{L} = 15\text{ms}$

The total elapsed time is less than 20ms , which is an acceptable value for most on-line implementations.



(a) Calculation times dependency on the number of vertices (b) Calculation times dependency on the number of variables

Figure 2: SGAM. Computational complexity analysis.

4 Case studies

In this section, SGAM has been applied in a synthetic process and in the TE process benchmark. To better illustrate the methodology, a comparison with the well known PCA monitoring technique is provided.

4.1 Five variables autocorrelated process

The first simulation process is the following multivariate open loop system, introduced by Ku and co-workers¹² and modified by Yu,²³

$$z(k) = \begin{bmatrix} 0.118 & -0.191 & 0.287 \\ 0.847 & 0.264 & 0.943 \\ -0.333 & 0.514 & -0.217 \end{bmatrix} z(k-1) + \begin{bmatrix} 1 & 2 \\ 3 & -4 \\ -2 & 1 \end{bmatrix} u(k-1) \quad (12)$$

$$y(k) = z(k) + v(k) \quad (13)$$

where u is the correlated input:

$$u(k) = \begin{bmatrix} 0.811 & -0.226 \\ 0.477 & 0.415 \end{bmatrix} u(k-1) + \begin{bmatrix} 0.193 & 0.689 \\ -0.320 & -0.749 \end{bmatrix} w(k-1) \quad (14)$$

and w is a two variables random vector with distribution $\mathcal{U}(-2, 2)$.

The measurement vector under analysis is defined as $x(k) = [y'(k), u'(k)] \in \mathbb{R}^5$. This is an interesting process, since it introduces dynamics and therefore autocorrelation into process variables.

The process was simulated for 300 samples, generating a normal (under control) dataset. The normal dataset was auto-scaled and used to implement PCA, three principal components has been selected to retain most ($\approx 90\%$) of the common cause variability. SGAM was also applied to this example using the procedure depicted on Table 1. Since data is auto-scaled (zero mean and unit variance), the neighborhood parameter was set to $\sigma = 1$, meaning that points at a distance $> 4\sigma$ will be almost disconnected (see Eq. 7).

The maximum response time of this system is about 15 samples. Therefore, n is set to 80, which is enough to have a graph that comprises the process dynamics and most of the common cause variability. Figures 3a and 3b show the resulting control charts for PCA and

SGAM under normal conditions and Figure 3c is a graphic representation of the adjacency matrix at sample 90 (i.e. it includes measurements from sample 11 to sample 90), the darkest the color the weaker the relation (similarity) between points.

To examine the monitoring performance, five disturbances of different type have been introduced in the input variable w_1 at sample 101 (see Table 3).

Table 3: Five variables autocorrelated process: Disturbances and faults introduced in w_1 at sample = 101.

Disturbance	Description
IDV(1)	Step of size = +3
IDV(2)	Drift fault, of slope = 0.03
IDV(3)	Noise increased by 50%
IDV(4)	Noise reduced by 20%
IDV(5)	Two consecutive steps, +3 at t=101 and -6 at t=201

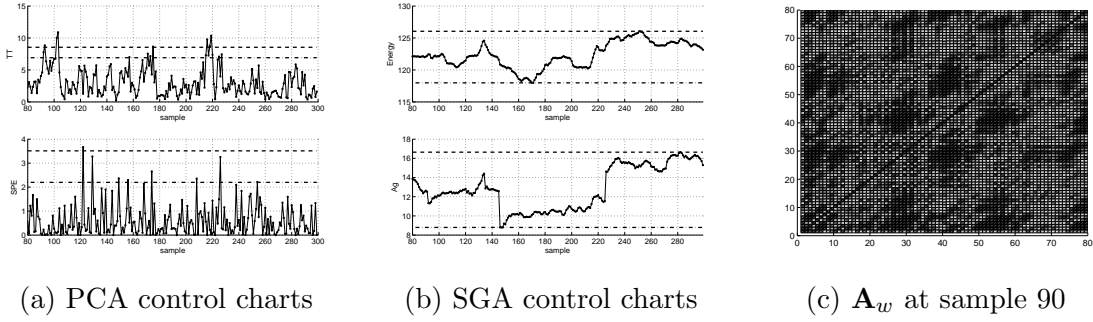


Figure 3: Five variables autocorrelated process. Normal state

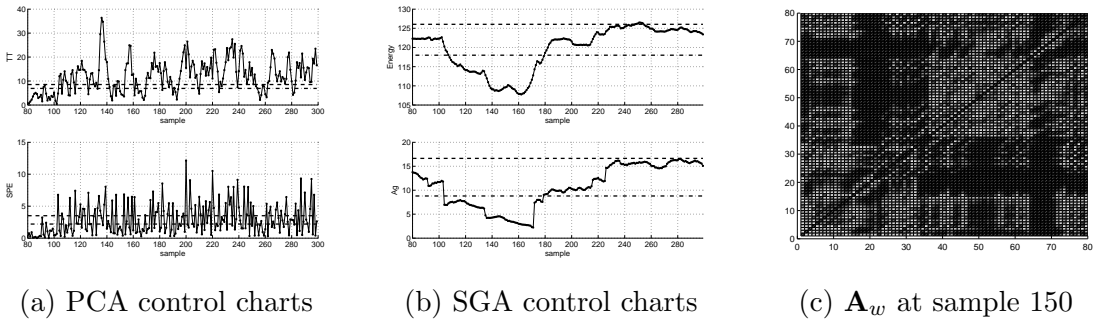


Figure 4: Five variables autocorrelated process. IDV(1) introduced at sample 101.

Figure 4 shows the monitoring results for the step disturbance IDV(1). PCA detects the fault at time 107 (three consecutive T^2 samples out of the 99% limit). The spectral detection

is produced at the same time, but it is much clearer, showing a decrement in the graph energy and in the algebraic connectivity. Note that after the step, the spectral features return to normal values, this is due to the fact that SGAM monitors local structures of data.

IDV(2) (see Figure 5) is a small drift disturbance, which is detected by the T^2 statistic intermittently at samples 138, 178, 185 and more clearly after sample 197. Using SGAM, the disturbance is distinctly detected in the *aG-control chart* from sample 137. Since the system does not return to a steady state, the spectral features does not completely return to the normality limits.

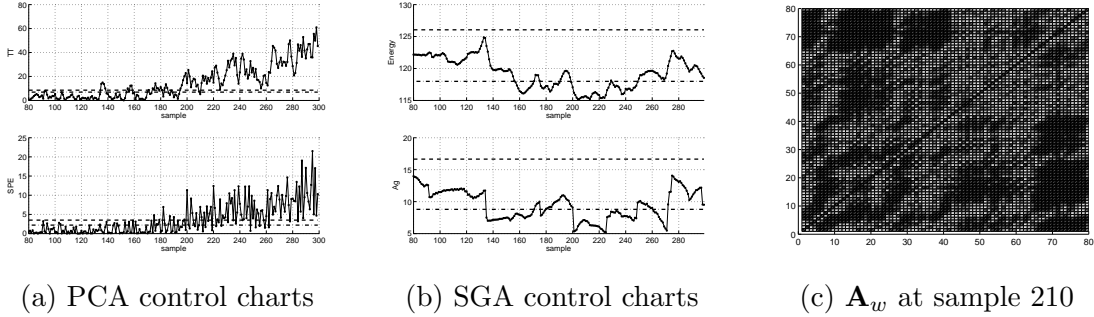


Figure 5: Five variables autocorrelated process. IDV(2) introduced at sample 101.

IDV(3) and IDV(4) (see Figures 6 and 7) correspond to increments and decrements in the noise levels respectively. IDV(3) is detected by PCA at sample 126, and again and with more clarity at sample 190. Detection using the *aG-control chart* is faster and sharper at sample 105. Note that since the noise level is kept higher than normal, the SGAM charts do not return to its normal values. Regarding IDV(4), although the reduction of noise causes a reduction in the values of its statistics (See Figure 7) this disturbance can not be detected by PCA. On the other hand, IDV(4) causes an increment in the graph energy (the graph vertices are more concentrated) that can be observed in *EG-control chart* from sample 132. The difference between the adjacency matrices in Figures 6c and 7c illustrate the changes in the graph vertices connectivity when changes in the noise levels occur.

As it was shown in Figure 4, after detection, if the system remains in an operating point with similar common cause variance the spectral features return to normal values. This gives

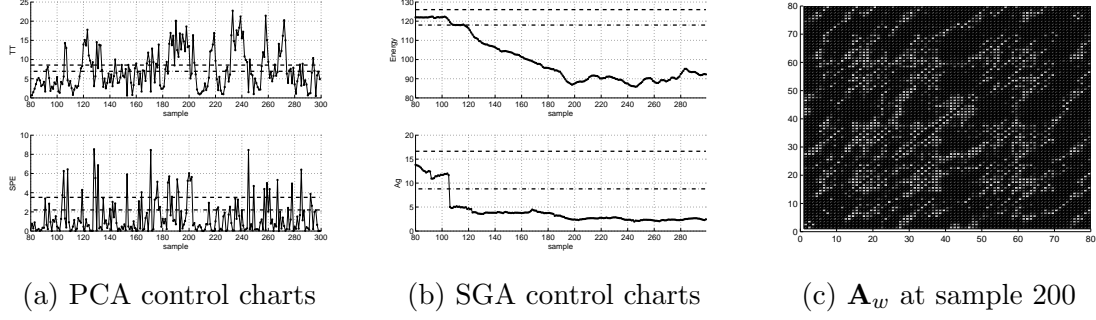


Figure 6: Five variables autocorrelated process. IDV(3) introduced at sample 101.

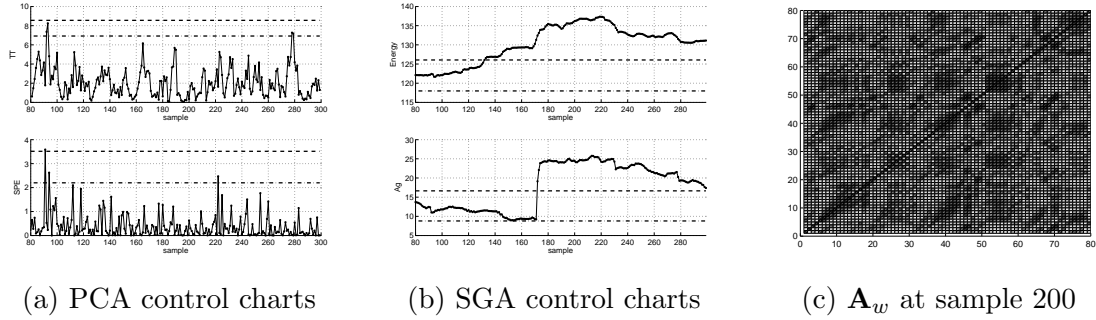


Figure 7: Five variables autocorrelated process. IDV(4) introduced at sample 101.

to the system some adaptive properties that are useful in consecutive faults detection such as IDV(5) (see Figure 8). Here a positive step is introduced at sample 101 and detected by PCA and SGAM statistics at sample 107. After some time (related to the time window size n), the SGAM statistics return to normal values. In this situation a new step is introduced at sample 201, which is clearly detected by SGAM charts (at sample 207). On the other hand, PCA statistics remain out of the limits since the detection of the first step at sample 107 and therefore the second disturbance is overlooked.

4.2 Tennessee Eastman Benchmark Process

In this case study the proposed approach is applied to the Tennessee Eastman (TE) Process.³⁹ The process has five main units (see Figure 9): an exothermic 2-phase reactor, a product condenser, a flash separator, a reboiled stripper, and a recycle compressor. The TE has 42 measured variables and 12 manipulated variables.

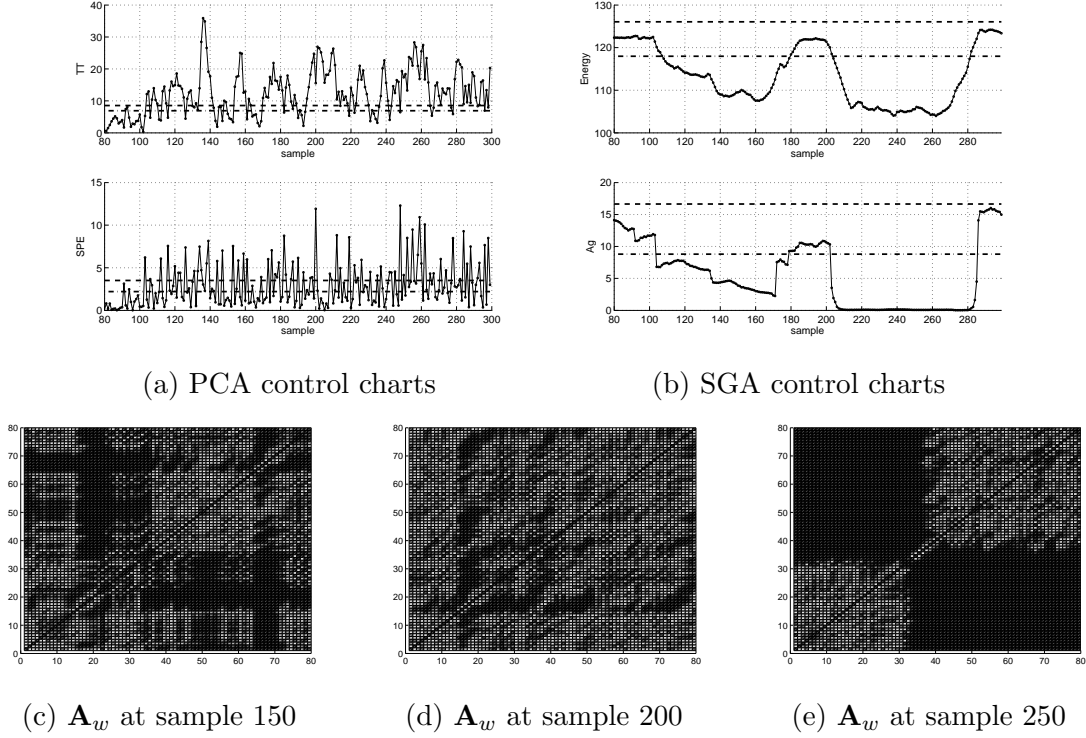


Figure 8: Five variables autocorrelated process. IDV(5) (Consecutive faults) introduced at sample 101.

Several control strategies have been applied to this benchmark. Since monitoring results change depending on the selected measurements, the control strategy and the sampled times, for sake of standardization in this work the data set published in⁴⁰ has been used. This dataset was obtained using the plant-wide control structure recommended in.⁴¹ It consists on 33 measured process variables collected with a 3 min sample time. All the disturbances proposed by Downs and Vogel³⁹ have been introduced at sample 161. Each dataset contains 960 samples. See Table 4 for detailed disturbance descriptions.

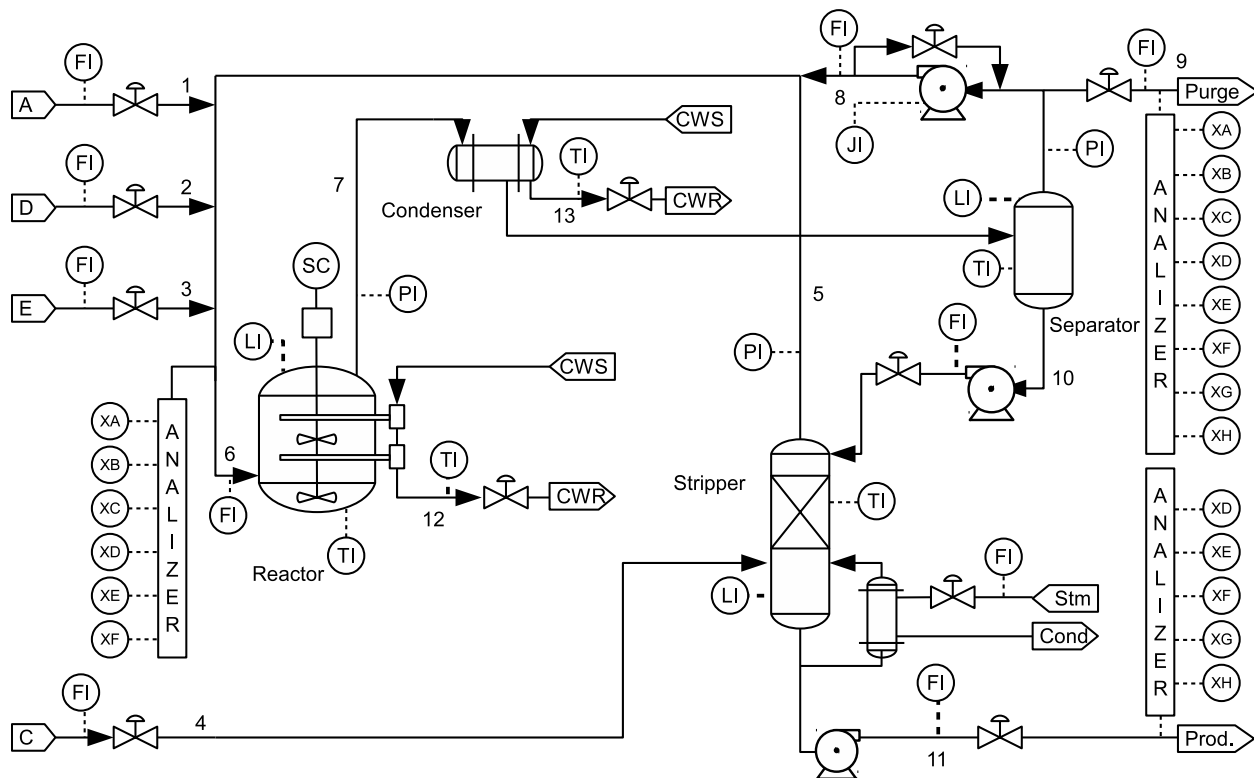
To better illustrate the obtained results, comparison with PCA and LPP monitoring is provided.

Using the auto-scaled normal dataset, SGAM has been applied with parameters $\sigma = 4$ (since Eq. 10 returned a value of 4.08) and $n = 100$, which is equivalent to 5 hours time windows.

A PCA model has been built using the same normal dataset, and 9 principal components

Table 4: Tennessee Eastman process: Disturbances and faults.

Disturbance	Description	Type
IDV(1)	A/C feed ratio, B composition constant (stream 4)	Step
IDV(2)	B composition, A/C feed ratio constant (stream 4)	Step
IDV(3)	D feed temperature (stream 2)	Step
IDV(4)	Reactor cooling water inlet temp.	Step
IDV(5)	Condenser cooling water inlet temp.	Step
IDV(6)	A feed loss (stream 1)	Step
IDV(7)	C header pressure loss-reduced availability (stream 4)	Step
IDV(8)	A, B, C feed composition (stream 4)	Random Variation
IDV(9)	D feed temperature (stream 2)	Random Variation
IDV(10)	C feed temperature (stream 4)	Random Variation
IDV(11)	Reactor cooling water inlet temperature	Random Variation
IDV(12)	Condenser cooling water inlet temperature	Random Variation
IDV(13)	Reaction kinetics	Slow Drift
IDV(14)	Reactor cooling water valve	Sticking
IDV(15)	Condenser cooling water valve	Sticking
IDV(16)	Unknown	Unknown
IDV(17)	Unknown	Unknown
IDV(18)	Unknown	Unknown
IDV(19)	Unknown	Unknown
IDV(20)	Unknown	Unknown
IDV(21)	The valve for Stream 4 was fixed at the steady state position	Constant Position



have been selected. LPP has been applied using $\sigma = 4$ as in SGAM and selecting 9 projected dimensions, as in PCA.

Faults alarms are considered to be triggered when three consecutive samples go over the confidence limit. To provide a fair comparison, confidence limits for all statistics has been set to avoid false alarms during the normal operation conditions.

As it has been explained before, SGAM is adaptive and, if the process stabilizes, its indicators return to normal values after detection. As a consequence, the direct application of reliability (i.e. defined as the percentage of points out of detection limits) measures can not be used to compare both techniques. Additionally, it is considered that detection times are more important than detection reliability since, once detected, a fault alarm can be easily hold on until the operator acknowledge the abnormal state. Therefore, detection time is chosen as the criteria to evaluate and compare monitoring performance; Table 5 shows the obtained results for PCA and SGAM.

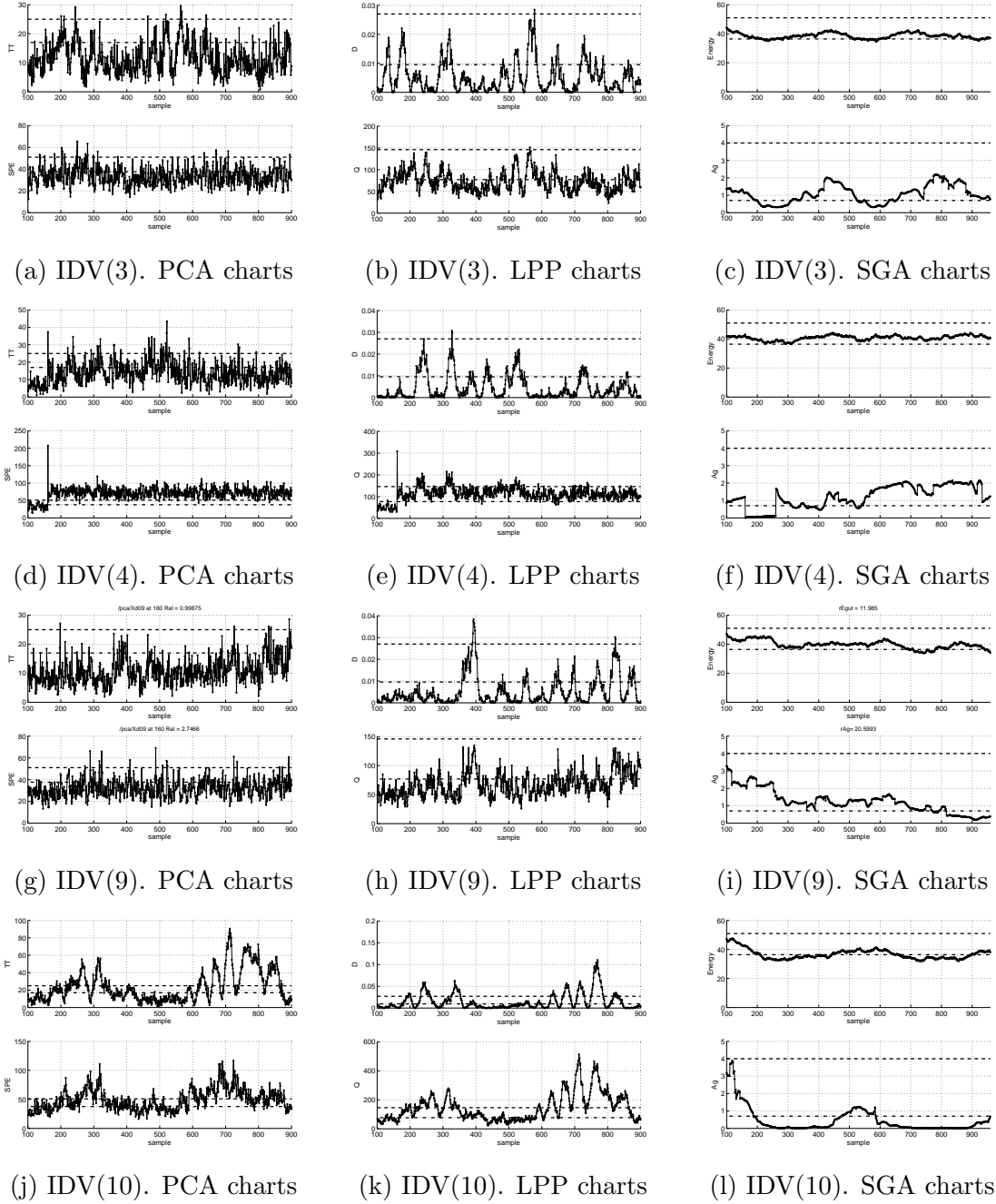


Figure 10: Tennessee Eastman Process. PCA, LPP and SGAM control charts for the Normal Operating Condition (NOC) and disturbances IDV(3), 4 and 10

It can be seen that most disturbances are of great magnitude and are easily detected by all PCA, LPP and SGAM at similar detection times (IDV1-IDV2, IDV5-IDV7, IDV12). IDV8, 17-18 are detected some samples later, but also with similar detection times for any technique.

With the exception of IDV9, which is particularly well detected by LPP (see Figure 10h), and IDV13 (see Figure 11d) , which is detected slightly faster by PCA; disturbances are better detected by SGAM. And even in the case of similar detection times, SGAM indicators tend to be more definitive. To illustrate this situation, PCA, LPP and SGAM control charts are presented for some of the most interesting disturbances.

Table 5: Tennessee Eastman process. Disturbance detection time comparison. Times are expressed in samples after fault incidence. BEST columns summarize the faster result for a given technique. To calculate detection time averages, undetected disturbances are penalized with 1000 samples.

	PCA			LPP			SGA		
	TT	SPE	BEST(TT,SPE)	D	Q	BEST(D,Q)	EG	aG	BEST(EG,aG)
IDV(1)	8	4	4	9	7	7	54	4	4
IDV(2)	18	13	13	32	16	16	48	14	14
IDV(3)	—	—	1000	—	—	1000	—	40	40
IDV(4)	—	4	4	—	62	62	—	2	2
IDV(5)	2	2	2	20	3	3	42	3	3
IDV(6)	10	2	2	32	2	2	53	2	2
IDV(7)	2	2	2	3	2	2	20	2	2
IDV(8)	24	21	21	31	24	24	58	22	22
IDV(9)	—	—	1000	231	—	231	—	611	611
IDV(10)	72	40	40	36	37	36	48	27	27
IDV(11)	13	12	12	304	12	12	26	6	6
IDV(12)	8	4	4	32	9	9	6	4	4
IDV(13)	50	38	38	51	48	48	70	46	46
IDV(14)	5	2	2	—	3	3	8	3	3
IDV(15)	676	597	597	628	632	628	592	582	582
IDV(16)	313	197	197	296	309	296	460	197	197
IDV(17)	30	26	26	62	28	28	52	26	26
IDV(18)	96	86	86	92	89	89	108	87	87
IDV(19)	—	132	132	—	—	1000	37	77	37
IDV(20)	87	84	84	87	86	86	130	84	84
IDV(21)	284	478	284	650	516	516	131	105	105
AVG	169			195			91		

Figures 10a, 10b and 10c show the IDV(3) PCA, LPP and SGAM control charts. PCA

and LPP statistics can not detect this disturbance with the selected set of sensors. IDV(3) is only detected by SGAM, in the form of small changes that can be seen in the *aG-control chart*. IDV(4) (see Figures 10d and 10f) is a step fault introduced in the reactor cooling water inlet temperature. Note that the fault is detected by both PCA and SGAM. In particular the *aG-control chart* shows a decrement in the algebraic connectivity, which returns to normal values when the control stabilizes the process. LPP detects this fault 60 samples later (see Figure 10e).

Random variations disturbances, such as IDV(10), IDV(11), are detected faster and with more clarity in SGAM control charts as it is made evident in Figures 10j, 10k and 10l and Figures 11a, 11b and 11c.

IDV(13) is the only case in which the fault is detected before using PCA (for just 8 samples). However, from Figures 11d, 11e and 11f it can be noted that the SGAM detection is much clearer as an abrupt reduction of the algebraic connectivity. IDV(15) (see Figures 11g, 11h and 11i) produces a slow drift that is detected by LPP at sample 628, by PCA at sample 758 and by SGAM 15 samples before. IDV(19) is also a random variation, which is easily detected by SGA but undetected by PCA (see Figures 12a and 12c). This disturbance is not detected by LPP (see Figure 12b). IDV(20) is detected by all techniques, but SGAM charts are more clear (see Figures 12d, 12e and 12f). Finally, IDV(21) is an unknown fault that primarily causes a reduction of process variability and a posterior drift in process variables. The reduction of process variability is detected by SGAM as an increment in the graph energy (see Figures 12g and 12i) at sample 266, much before the occurrence of the process variables drift. PCA only detects this disturbance after the drift at sample 639.

SGAM is the only technique that can detect all the presented faults. In addition, it provides good all around results, as shown by the average detection time (see Table 5).

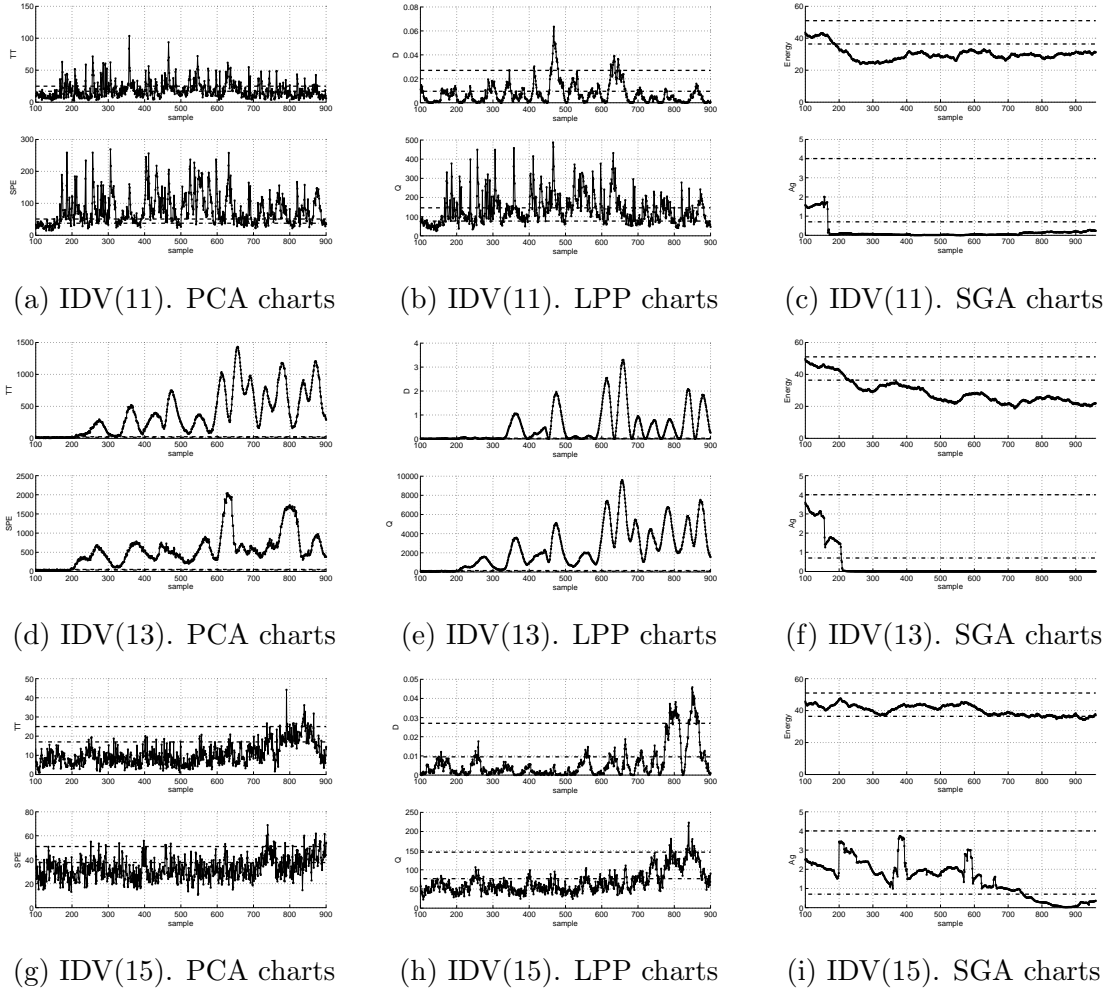
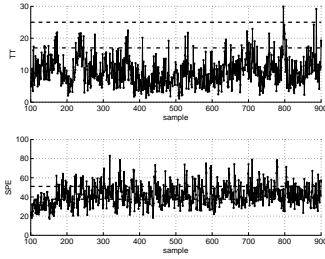
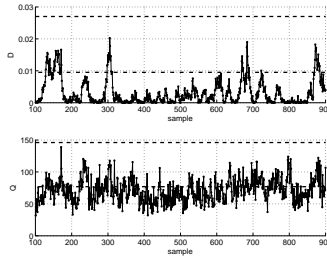


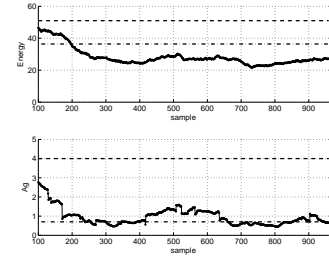
Figure 11: Tennessee Eastman Process. PCA, LPP and SGAM control charts for the Normal Operating Condition (NOC) and disturbances IDV(11), 12, 14 and 15



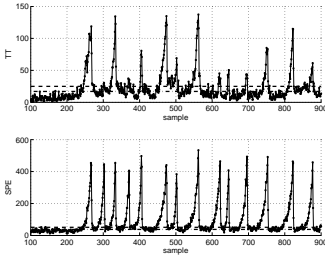
(a) IDV(19). PCA charts



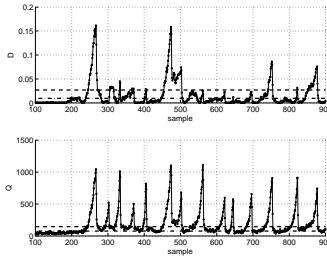
(b) IDV(19). LPP charts



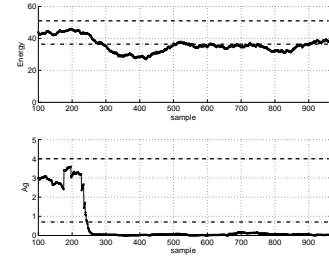
(c) IDV(19). SGA charts



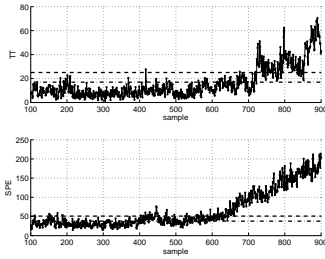
(d) IDV(20). PCA charts



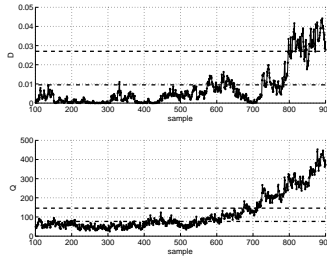
(e) IDV(20). LPP charts



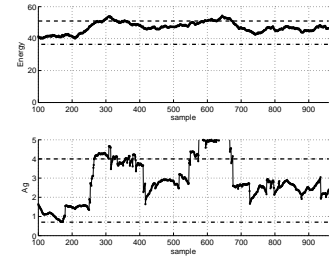
(f) IDV(20). SGA charts



(g) IDV(21). PCA charts



(h) IDV(21). LPP charts



(i) IDV(21). SGA charts

Figure 12: Tennessee Eastman Process. PCA, LPP and SGAM control charts for IDV(19), 20 and 21

5 Conclusions and future work

In this work Spectral Graph Analysis Monitoring (SGAM) has been presented. SGAM is a new process monitoring technique, which is not based in a reduced space model.

SGAM has been illustrated in autocorrelated and non-linear synthetic cases, and applied to the well known Tennessee Eastman process benchmark. Several types of process disturbances have been evaluated, including steps, drifts and random variations.

The technique shows promising results to identify abnormal events. It can be used as an alternative or complement to PCA, LPP and other on-line statistical monitoring approaches. Since statistics sometimes adapt to the new faulty state, a simple hold on alarm management scheme would improve the obtained results. In such systems, alarms stay active (even if the fault indicator returns to the normal state) until the operator acknowledge the fault.

As a future work, the intrinsic adaptive capabilities of SGAM will be further investigated. Also, it will be extended, exploring its potentialities as a fault diagnosis tool.

Acknowledgement

Financial support from CONICET, ANPCyT and FCEIA-UNR is fully acknowledged. I would also like to thank Dr. David Zumoffen for the many insightful discussions during the development of this work.

References

1. Venkatasubramanian, V.; Rengaswamy, R.; Yin, K.; Kavuri, S. N. *Computers & Chemical Engineering* **2003**, *27*, 293–311.
2. Venkatasubramanian, V.; Rengaswamy, R.; Kavuri, S. N. *Computers & Chemical Engineering* **2003**, *27*, 313–326.
3. Venkatasubramanian, V.; Rengaswamy, R.; Kavuri, S. N.; Yin, K. *Computers & Chemical Engineering* **2003**, *27*, 327–346.
4. Musulin, E.; Roda, F.; Basualdo, M. *Computers & Chemical Engineering* **2013**, *59*, 164–177.
5. Kano, M.; Tanaka, S.; Hasebe, S.; Hashimoto, I.; Ohno, H. *AIChE Journal* **2003**, *49*, 969–976.
6. Ge, Z.; Song, Z. *Industrial & Engineering Chemistry Research* **2007**, *46*, 2054–2063.
7. Qin, S. J. *Annual Reviews in Control* **2012**, *36*, 220–234.
8. Xie, X.; Shi, H. *Industrial & Engineering Chemistry Research* **2012**, *51*, 5497–5505.
9. Tong, C.; Yan, X. *Chemometrics and Intelligent Laboratory Systems* **2014**, *130*, 20–28.
10. Jackson, J. *A user's guide to principal components*; John Wiley: New York, 1991.
11. Nomikos, P.; MacGregor, J. *Technometrics* **1995**, *37*, 41–59.
12. Ku, W.; Storer, R. H.; Georgakis, C. *Chemometrics and Intelligent Laboratory Systems* **1995**, *30*, 179–196.
13. Musulin, E.; Yelamos, I.; Puigjaner, L. *Industrial & Engineering Chemistry Research* **2006**, *45*, 1739–1750.
14. Ge, Z.; Yang, C.; Song, Z. *Chemical Engineering Science* **2009**, *64*, 2245–2255.

15. Bakshi, B. *AIChE Journal* **1998**, *44*, 1596–1610.
16. Tenenbaum, J. B.; De Silva, V.; Langford, J. C. *Science* **2000**, *290*, 2319–2323.
17. Roweis, S. T.; Saul, L. K. *Science* **2000**, *290*, 2323–2326.
18. Belkin, M.; Niyogi, P. *Neural computation* **2003**, *15*, 1373–1396.
19. Niyogi, X. Locality preserving projections. In *Neural information processing systems* **2004**, *16*, 153–160
20. Von Luxburg, U. *Statistics and computing* **2007**, *17*, 395–416.
21. Van der Maaten, L.; Postma, E.; Van Den Herik, H. *Journal of Machine Learning Research* **2009**, *10*, 1–41.
22. Zhang, M.; Ge, Z.; Song, Z.; Fu, R. *Industrial & Engineering Chemistry Research* **2011**, *50*, 6837–6848.
23. Yu, J. *Journal of Process Control* **2012**, *22*, 1358–1373.
24. Bondy, J.; Murty, S. *Graph theory with applications*; Elsevier Science Publishing Co., Inc.: 5 ed.; 1985.
25. Nikiforov, V. *Journal of Mathematical Analysis and Applications* **2007**, *326*, 1472–1475.
26. Gutman, I.; Shao, J. *Linear Algebra and its Applications* **2011**, *435*, 2425–2431.
27. Cvetković, D. M.; Rowlinson, P.; Simić, S. *An introduction to the theory of graph spectra*; Cambridge University Press Cambridge: 2010.
28. Cvetković, D.; Simić, S. *Linear Algebra and its Applications* **2011**, *434*, 1545–1562.
29. Castaño Arranz, M.; Birk, W. *Journal of Process Control* **2012**, *22*, 280–295.
30. Yang, F.; Shah, S.; Xiao, D. *International journal of applied mathematics and computer science* **2012**, *22*, 41–53.

31. Gutierrez-Perez, J.; Herrera, M.; Perez-Garcia, R.; Ramos-Martinez, E. *Mathematical and Computer Modelling* **2013**, *57*, 1853–1859
32. Zumoffen, D.; Musulin, E. *Computers & Chemical Engineering* **2013**, *56*, 80–88.
33. Gutman, I.; Li, X.; Zhang, J. Analysis of Complex Networks. From Biology to Linguistics. In ; Dehmer, M.; Emmert-Streib, F., Eds.; Wiley-VCH: 2009; Chapter Graph Energy, pages 145–174.
34. Gutman, I. *J. Serb. Chem. Soc.* **2005**, 441–456.
35. Nikiforov, V. *Journal of Mathematical Analysis and Applications* **2007**, *326*, 1472–1475.
36. Mohar, B. *Graphs and combinatorics* **1991**, *7*, 53–64.
37. Mohar, B. *Journal of Combinatorial Theory, Series B* **1989**, *47*, 274–291.
38. Trefethen, L. N.; Bau III, D. *Numerical linear algebra*; Society for Industrial and Applied Mathematics (SIAM): Philadelphia, 1997.
39. Downs, J.; Vogel, E. *Computers & Chemical Engineering* **1993**, *17*, 245–255.
40. Braatz, R. D. “Tennessee Eastman Problem Simulation Data”, <http://web.mit.edu/braatzgroup/links.html>, Accessed: 2013-06-30.
41. Lyman, P.; Georgakis, C. *Computers & Chemical Engineering* **1995**, *19*, 321–331.

Nomenclature

A	Graph adjacency matrix
A_w	Graph weighted adjacency matrix
<i>a</i>	Adjacency matrix element
<i>a(G)</i>	Graph algebraic connectivity
D	Diagonal matrix of vertex degrees
<i>e</i>	Graph edge
$\mathbb{E}(G)$	Graph Energy
<i>G</i>	Graph
<i>i(G)</i>	Graph isoperimetric number
<i>K</i>	Number of time windows in a dataset
<i>k</i>	Sample time
L	Discrete Laplacian matrix
<i>m</i>	Number of variables
<i>n</i>	Number of graph vertices
<i>R_N</i>	normal dataset radio
<i>TW(k)</i>	Moving time window
<i>V</i>	Graph vertices
<i>w</i>	Adjacency matrix weights
X	Measurement data set
x_m	Measurement vector

Greek Letters

λ	Adjacency matrix eigenvalues
-----------	------------------------------

Subindexes

<i>i, j</i>	Graph vertices
-------------	----------------

Acronyms

GLPCA	Global Local Structure Analysis
LGPCA	Local and Global PCA
LPP	Locality Preserving Projections
NOC	Normal Operation Condition
PC	Principal Component
PCA	Principal Component Analysis
SGAM	Spectral Graph Analysis Monitoring